



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/775,982

02/10/2004

Jose A. Blakeley

MSFT-3027

1307

41505

7590

08/20/2007

WOODCOCK WASHBURN LLP (MICROSOFT CORPORATION)

CIRA CENTRE, 12TH FLOOR

2929 ARCH STREET

PHILADELPHIA, PA 19104-2891

EXAMINER

MORRISON, JAY A

ART UNIT

PAPER NUMBER

2168

MAIL DATE

DELIVERY MODE

08/20/2007

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

10/775,982

Applicant(s)

BLAKELEY ET AL.

Examiner

Jay A. Morrison

Art Unit

2168

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 29 May 2007.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-27 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-27 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 5/29/2007 has been entered.

Claim Rejections - 35 USC § 102

2. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

3. Claims 1,4,8-9,14,17,21-22 and 27 are rejected under 35 U.S.C. 102(b) as being anticipated by Schallehn et al. ('Schallehn' hereinafter) "Advanced Grouping and Aggregation for Data Integration", by Schellehn, E., Kai-Uwe, S., and Saake, G., Conference on Information and Knowledge Management archive, Proceedings of the tenth international conference on Information and knowledge management, Atlanta, Georgia, USA, Pages: 547 - 549, Year: 2001, ISBN:1-58113-436-3).

As per claim 1, Schallehn teaches

A method for allowing users to define new aggregates in a database system,
comprising: (see abstract)

receiving code that instantiates an object of a class defining the structure of a
user-defined aggregate and methods (interface, page 548, column 2, paragraphs 4-6)

that can be invoked on instances of the user-defined aggregate; ('can be'
denotes an optionally recited limitation and optionally recited limitations are not
guaranteed to take place and are therefore not required to be taught, see MPEP § 2106
Section II(C))

and enforcing a contract against the class so that the code, when executed at
runtime, satisfies requirements of the contract to ensure desired execution thereof, the
contract requiring that the class comprise a first method (interface, page 548, column 2,
paragraphs 4-6)

that is invoked to initialize the computation of an instance of the user-defined
aggregate, (init() in interface UDG, page 548, second column, 4-5 paragraph)

a second method (interface, page 548, column 2, paragraphs 4-6)

that is invoked to accumulate a plurality of values to be aggregated, (aggregation
in UDG, page 548, second column 4-5 paragraph)

and a third method (interface, page 548, column 2, paragraphs 4-6)

that is invoked to compute a final result of the instance of the user-defined
aggregate. (finish() in interface UDG, page 548, second column 4-5 paragraphs)

As per claim 4,
the contract further comprises a requirement that the class comprise a public constructor having no arguments. (page 548, column 2, paragraphs 2-4)

As per claim 8,
the code that implements the class comprises managed code. (java, page 548, column 2, second paragraph)

As per claim 9,
receiving a query that requires evaluation of the user-defined aggregate over a group of elements; (page 549, first column, paragraphs 1-3)
instantiating an instance of the class that defines the structure of the user-defined aggregate; (page 549, first column, paragraphs 1-3)
invoking said first method to initializing the instance of the user-defined aggregate; (page 549, first column, paragraphs 1-3)
for each element of the group, invoking said second method on the instance of the user-defined aggregate to accumulate each element; (page 549, first column, paragraphs 1-3)
and invoking said fourth method to return a value of the user-defined aggregate. (page 549, first column, paragraphs 1-3)

As per claims 14,17,21-22

These claims are rejected on grounds corresponding to the arguments given above for rejected claims 2,4,8-9, respectively, and are similarly rejected.

As per claim 27, Schallehn teaches

A computer-readable medium having program code stored thereon for allowing users to define new aggregates in a database system, the program code, when executed by a computer, causing the computer to perform the method recited in claim 1.
(see abstract)

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to

consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

5. Claims 2 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Schallehn et al. ('Schallehn' hereinafter) "Advanced Grouping and Aggregation for Data Integration", by Schellehn, E., Kai-Uwe, S., and Saake, G., Conference on Information and Knowledge Management archive, Proceedings of the tenth international conference on Information and knowledge management, Atlanta, Georgia, USA, Pages: 547 - 549, Year: 2001, ISBN:1-58113-436-3) in view of Barry et al. ('Barry' hereinafter) ("Solving the Java object storage problem", by Douglas Barry and Torsten Stanienda, Computer, Volume 31, Issue 11, Nov. 1998, Page(s):33 – 40, IEEE).

As per claim 2,

Schallen does not explicitly indicate "the contract further comprises a requirement that the class specify one of a plurality of different formats which describes how instances of the user-defined aggregate are to be persisted in a database store".

However, Barry discloses "the contract further comprises a requirement that the class specify one of a plurality of different formats which describes how instances of the user-defined aggregate are to be persisted in a database store" (code required to implement class using ODMG binding which maps Java objects directly between database and object cache, page 34, second column, ODMG Binding for Java, page 37, second column, second paragraph; implements PersistenceCapable, figure 3).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Schallen and Barry because using the steps of "the contract further comprises a requirement that the class specify one of a plurality of different formats which describes how instances of the user-defined aggregate are to be persisted in a database store" would have given those skilled in the art the tools to improve the invention by allowing objects to be permanently stored. This gives the user the advantage of not having to rebuild objects.

As per claim 15,

This claim is rejected on grounds corresponding to the arguments given above for rejected claim 2 and is similarly rejected.

6. Claims 3,12-13,16 and 25-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Schallehn et al. ('Schallehn' hereinafter) "Advanced Grouping and Aggregation for Data Integration", by Schellehn, E., Kai-Uwe, S., and Saake, G., Conference on Information and Knowledge Management archive, Proceedings of the tenth international conference on Information and knowledge management, Atlanta, Georgia, USA, Pages: 547 - 549, Year: 2001, ISBN:1-58113-436-3) in view of Soloviev ("An overview of three commercial object-oriented database management systems: ONTOS, ObjectStore, and O2", by Valery Soloviev, March 1992, ACM SIGMOD Record, Volume 21 Issue 1, pages 93-104, Publisher: ACM Press).

Art Unit: 2168

As per claim 3,

the contract further comprises a requirement that the class comprise a fourth method (page 548, column 2, paragraphs 4-6)

Schallehn does not explicitly indicate “that is invoked to merge an instance of the user-defined aggregate with another partial aggregation”.

However, Soloviev discloses “that is invoked to merge an instance of the user-defined aggregate with another partial aggregation” (concatenation, append, page 100, second column, bullet points).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Schallehn and Soloviev because using the steps of “that is invoked to merge an instance of the user-defined aggregate with another partial aggregation” would have given those skilled in the art the tools to improve the invention by adding more functionality to existing applications. This gives the user the advantage of having more options for extending capabilities.

As per claim 12,

Schallehn does not explicitly indicate “the class defining the user-defined aggregate further comprises one of: an attribute that indicates whether the user-defined aggregate is invariant to duplicates; an attribute that indicates whether the user-defined aggregate is invariant to NULL values; an attribute that indicates whether the user-defined aggregate is invariant to order; and an attribute that indicates whether the user-

Art Unit: 2168

defined aggregate returns NULL if the group on which the user-defined aggregate is to be computed is empty”.

Soloviev discloses “the class defining the user-defined aggregate further comprises one of: an attribute that indicates whether the user-defined aggregate is invariant to duplicates; an attribute that indicates whether the user-defined aggregate is invariant to NULL values; an attribute that indicates whether the user-defined aggregate is invariant to order; and an attribute that indicates whether the user-defined aggregate returns NULL if the group on which the user-defined aggregate is to be computed is empty” (page 100, first column, 4th paragraph).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Schallehn and Soloviev because using the steps of “the class defining the user-defined aggregate further comprises one of: an attribute that indicates whether the user-defined aggregate is invariant to duplicates; an attribute that indicates whether the user-defined aggregate is invariant to NULL values; an attribute that indicates whether the user-defined aggregate is invariant to order; and an attribute that indicates whether the user-defined aggregate returns NULL if the group on which the user-defined aggregate is to be computed is empty” would have given those skilled in the art the tools to improve the invention by allowing membership to be tested and constrained. This gives the user the advantage of having rules predefined.

As per claim 13,

Schallehn does not explicitly indicate "determining a method of computation of an instance of the user-defined aggregate based at least in part upon a value of said one of said attributes".

However, Soloviev discloses "determining a method of computation of an instance of the user-defined aggregate based at least in part upon a value of said one of said attributes" (page 100, first column, 4th paragraph).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Schallehn and Soloviev because using the steps of "determining a method of computation of an instance of the user-defined aggregate based at least in part upon a value of said one of said attributes" would have given those skilled in the art the tools to improve the invention by allowing membership to be tested and constrained. This gives the user the advantage of having rules predefined.

As per claims 16, 25-26,

These claims are rejected on grounds corresponding to the arguments given above for rejected claims 3 and 12-13, respectively, and are similarly rejected.

7. Claims 5 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Schallehn et al. ('Schallehn' hereinafter) "Advanced Grouping and Aggregation for Data Integration", by Schellehn, E., Kai-Uwe, S., and Saake, G., Conference on Information and Knowledge Management archive, Proceedings of the tenth international conference on Information and knowledge management, Atlanta, Georgia, USA, Pages: 547 - 549,

Year: 2001, ISBN:1-58113-436-3) in view of Kriens et al. ('Kriens' hereinafter) (Patent Number 5,864,862).

As per claim 5,

Schallehn does not explicitly indicate "storing metadata about the user-defined aggregate".

However, Kriens discloses "storing metadata about the user-defined aggregate" (column 22, lines 21-67).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Schallehn and Kriens because using the steps of "storing metadata about the user-defined aggregate" would have given those skilled in the art the tools to improve the invention by allowing their own structure to be reflected on during run-time. This gives the user the advantage of have meta-information about objects available at run-time.

"for subsequent use by the database system in creating instances of the user-defined aggregate". ('for' indicates intended use; *Minton v. Nat 'l Ass 'n of Securities Dealers, Inc.*, 336 F.3d 1373, 1381, 67 USPQ2d 1614, 1620 (Fed. Cir. 2003) "whereby clause in a method claim is not given weight when it simply expresses the intended result of a process step positively recited." Examples of claim language, although not exhaustive, that may raise a question as to the limiting effect of the language in a claim are: (A) "adapted to" or "adapted for " clauses; (B) "wherein" clauses; and (C)

"whereby" clauses. Therefore intended use limitations are not required to be taught, see MPEP § 2106 Section II(C), MPEP 2111.04 [R-3])

As per claim 18,

This claim is rejected on grounds corresponding to the arguments given above for rejected claim 5 and is similarly rejected.

8. Claims 6-7,19-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Schallehn et al. ('Schallehn' hereinafter) "Advanced Grouping and Aggregation for Data Integration", by Schellehn, E., Kai-Uwe, S., and Saake, G., Conference on Information and Knowledge Management archive, Proceedings of the tenth international conference on Information and knowledge management, Atlanta, Georgia, USA, Pages: 547 - 549, Year: 2001, ISBN:1-58113-436-3) in view of Barry et al. ('Barry' hereinafter) ("Solving the Java object storage problem", by Douglas Barry and Torsten Stanienda, Computer, Volume 31, Issue 11, Nov. 1998, Page(s):33 – 40, IEEE) and further in view of Berg ("How Do I Create Persistent Java Objects", Dr. Dobbs Journal, April 1997, pages 98-101).

As per claim 6, Schallehn teaches

Neither Schallehn nor Barry explicitly indicate "comprises: a first format in which an instance of the user-defined aggregate is automatically serialized in accordance with

a native format of the database system; and a second format in which an instance of the user-defined aggregate is serialized in a manner defined by the class.”

However, Berg discloses “comprises: a first format in which an instance of the user-defined aggregate is automatically serialized in accordance with a native format of the database system; and a second format in which an instance of the user-defined aggregate is serialized in a manner defined by the class” (page 98, second column, second paragraph through the third column, first paragraph).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Schallehn, Berry and Berg because using the steps of “comprises: a first format in which an instance of the user-defined aggregate is automatically serialized in accordance with a native format of the database system; and a second format in which an instance of the user-defined aggregate is serialized in a manner defined by the class” would have given those skilled in the art the tools to improve the invention by allowing the database to persist the data. This gives the user the advantage of not having to generate the data each time it is required.

As per claim 7,

Neither Schallehn nor Barry explicitly indicate “further comprises a third format in which an instance of the user-defined aggregate is serialized in accordance with a method provided by a managed code programming module.”

However, Berg discloses "further comprises a third format in which an instance of the user-defined aggregate is serialized in accordance with a method provided by a managed code programming module" (JDK, page 98, second column, third paragraph).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Schallehn, Berry and Berg because using the steps of "further comprises a third format in which an instance of the user-defined aggregate is serialized in accordance with a method provided by a managed code programming module" would have given those skilled in the art the tools to improve the invention by allowing the database to persist the data. This gives the user the advantage of not having to generate the data each time it is required.

As per claims 19-20

These claims are rejected on grounds corresponding to the arguments given above for rejected claims 6-7, respectively, and are similarly rejected.

9. Claims 10-11,23-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Schallehn et al. ('Schallehn' hereinafter) "Advanced Grouping and Aggregation for Data Integration", by Schellehn, E., Kai-Uwe, S., and Saake, G., Conference on Information and Knowledge Management archive, Proceedings of the tenth international conference on Information and knowledge management, Atlanta, Georgia, USA, Pages: 547 - 549, Year: 2001, ISBN:1-58113-436-3) in view of Berg ("How Do I Create Persistent Java Objects", Dr. Dobbs Journal, April 1997, pages 98-101).

As per claim 10,

Schallehn does not explicitly indicate “deserializing the instance of the user-defined aggregate prior to invoking said second method for an element of the group and then serializing the instance after invoking said second method for the element.”

However, Berg discloses “deserializing the instance of the user-defined aggregate prior to invoking said second method for an element of the group and then serializing the instance after invoking said second method for the element” (page 98, second column, second paragraph through the third column, first paragraph).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Schallehn and Berg because using the steps of “deserializing the instance of the user-defined aggregate prior to invoking said second method for an element of the group and then serializing the instance after invoking said second method for the element” would have given those skilled in the art the tools to improve the invention by allowing the database to persist the data. This gives the user the advantage of not having to generate the data each time it is required.

As per claim 11,

Schallehn does not explicitly indicate “deserializing the instance of the user-defined aggregate prior to invoking said second method for a first element of the group; caching the deserialized instance of the user-defined aggregate; for each element of the

group, invoking said second method on the cached instance of the user-defined type; and, thereafter, serializing the cached instance of the user-defined type.”

However, Berg discloses “deserializing the instance of the user-defined aggregate prior to invoking said second method for a first element of the group; caching the deserialized instance of the user-defined aggregate; for each element of the group, invoking said second method on the cached instance of the user-defined type; and, thereafter, serializing the cached instance of the user-defined type” (page 98, second column, second paragraph through the third column, first paragraph).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Schallehn and Berg because using the steps of “deserializing the instance of the user-defined aggregate prior to invoking said second method for a first element of the group; caching the deserialized instance of the user-defined aggregate; for each element of the group, invoking said second method on the cached instance of the user-defined type; and, thereafter, serializing the cached instance of the user-defined type” would have given those skilled in the art the tools to improve the invention by allowing the database to persist the data. This gives the user the advantage of not having to generate the data each time it is required.

As per claims 23-24

These claims are rejected on grounds corresponding to the arguments given above for rejected claims 10-11, respectively, and are similarly rejected.

Response to Arguments

10. As per claim 1 and 14, Applicant's arguments filed 4/27/2007 have been fully considered but they are not persuasive. Applicant argues that Schallehn describes a class for creating user-defined grouping and does not define a user-defined aggregate. Respectfully, Schallehn discloses not a class but an interface for both a user-defined aggregate (page 548, 2nd paragraph, "public interface UDA") as well as a user-defined grouping (page 548, 5th paragraph, "public interface UDG"). Further, according to the specification of the instant application an aggregate does not have to be limited to a simple summing (see instant application, figures 2A and 2B). While figure 2B of the instant application describes a summation-type aggregate in a class that fulfills the contract of an the claimed user-defined aggregate (i.e. implements the interface), figure 2A shows that the interface is much more broad and can be fulfilled by a class which aggregates via appending values in order to make a list, set, or some other structure. In fact the claimed interface is very broad because the class implementing it can do most anything as long as it meets the relatively simple interface contract. It is therefore respectfully submitted that Schallehn does in fact disclose the limitation.

11. Applicant's arguments with respect to claims 2-13 and 15-27 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

12. The prior art made of record, listed on form PTO-892, and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jay A. Morrison whose telephone number is (571) 272-7112. The examiner can normally be reached on M-F 8-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tim Vo can be reached on (571) 272-3642. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).



TIM VO
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Jay Morrison
TC2100

Tim Vo
TC2100